“AZƏRBAYCAN HAVA YOLLARI” CJSC NATIONAL AVIATION ACADEMY”

**Topic:** Cronjobs in Linux

**Subject:** Operating System

**Teacher:** Sahmaliyev Memmed

**Student:** Aliyeva Qumru

**Group:** 2450-I

**Date:** 22.10.22

**Signature:**

**Baku 2022**

**Cronjobs in Linux**

**What are cron, cron job, and crontab?**

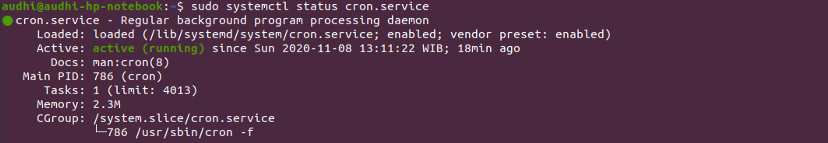
Cron is a system that helps Linux users to schedule any task. However, a cron job is any defined task to run in a given time period. It can be a shell script or a simple bash command. Cron job helps us automate our routine tasks, it can be hourly, daily, monthly, etc.

Note: in most of Linux system, we must get a permission of system administrator before defining a spesific cron job that is listed on crontab

Meanwhile, the crontab stands for cron table. It is a Linux system file that contains a list of the cron job. We define our task — bash command, shell script, Python script, etc scheduled in crontab.

# Check cron service on Linux system

sudo systemctl status cron.service



Understand a cron job syntax

It’s time to learn about cron job syntax on crontab.

crontab -a <filename>: create a new <filename> as crontab file

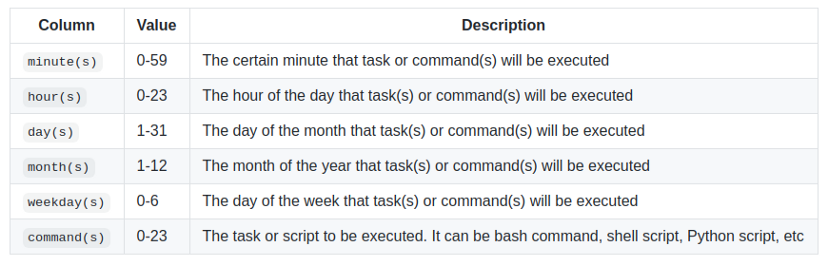
crontab -e: edit our crontab file or create one if it doesn’t already exist

crontab -l: show up our crontab file

crontab -r: delete our crontab file

crontab -v: show up the last time we have edited our crontab file

minute(s) hour(s) day(s) month(s) weekday(s) command(s)



How to handle an error on your cron job

If the cron job encounters an error, the default, it will send an email to the system administrator. Instead, we will find out two common ways when we encounter the error.

1 Send output to a specific file

It’s a common way and I always use it on my cron job. It’s simple. We just need to create a file that will save our cron job logs. It will print out the output just in case the job is accomplished properly or print out an error if it fails.

In this tutorial, it just created a log.out file. The output will be redirected to log.out.

\* \* \* \* \* cd /home/audhi && /bin/bash shell-script.sh >> log.out

The description of the above syntax on a crontab file is as follows.

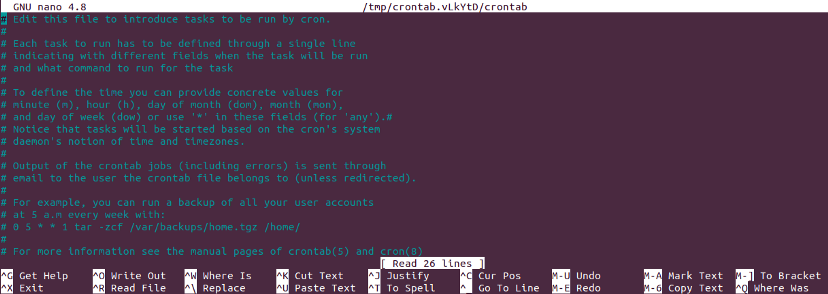
The \* \* \* \* \* means that a task will be executed every minute of every hour of every day of every month and every day of the week

The directory will be switched to /home/audhi where the shell-script.sh is located

/bin/bash is the path and executable of the Bash shell

The >> symbol will append the output to an existing file (log.out), while a single > symbol will overwrite the file

The shell-script.sh is a certain shell script



2 Use /dev/null

We can easily send our cron job logs and error to the dev/null instead of an alert via email. Whatever we send or write to dev/null, it will be discarded.

\* \* \* \* \* cd /home/audhi && /bin/bash shell-script.sh > /dev/null 2>&1

A little description of the commands:

The > /dev/null tells the cron to redirect the output (STDOUT) to /dev/null

2 is the file descriptor for Standard Error (STDERR)

& is the symbol for file descriptor (without it, the following 1 will be a filename)

1 is the file descriptor for Standard Out (STDOUT)

Note: The 2>&1 tells the cron to redirect all errors (STDERR) to same as standard out (STDOUT)

Write a simple cron automation script

To complete this article, I have created a Python script to demonstrate how to use a cron job. This Python script will collect the Covid-19 data from one of the largest Indonesian online news, Kompas News. You can find out my Python script for Covid-19 data web scraping at my GitHub repo. Its filename is Web Scraping Covid-19 Kompas News.py.

Open our terminal and type crontab -e to open a crontab file. Then, scroll down and type the following command.

5 16 \* \* \* cd 'covid19 data' && /usr/bin/python3 'Web Scraping Covid-19 Kompas News.py' >> test.out

The description of the above syntax on a crontab file is as follows.

The crontab is located in /home and my script is in /home/covid19 data, so we need to switch to the /home/covid19 data first

The python3 interpreter is located in /usr/bin/python3

The output will be redirected to test.out file in /home/covid19 data